Package ‘SurrogateParadoxTest’

January 20, 2025

Type Package

Title Empirical Testing of Surrogate Paradox Assumptions

Version 2.0

Date 2025-01-20

Author Emily Hsiao

Maintainer Emily Hsiao <ehsiao@utexas.edu>

Description Provides functions to nonparametrically assess assumptions necessary to prevent the surrogate paradox through hypothesis tests of stochastic dominance, monotonicity of regression functions, and non-negative residual treatment effects. More details are available in Hsiao et al 2025 (under review). A tutorial for this package can be found at <https://laylaparast.com/home/SurrogateParadoxTest.html>.

License GPL

Imports stats, MonotonicityTest

NeedsCompilation no

Contents

a_b_hat *Helper function*

Description

Helper function for monotonicity test

Usage

```
a_b_hat(r, s, X, Y)
```

Arguments

r	Index to start summation.
s	Index to end summation.
X	Vector of X values.
Y	Vector of Y values.

Value

a_hat	Numeric value of a_hat
b_hat	Numeric value of b_hat

Author(s)

Emily Hsiao

barrett_donald_cutoff *Helper function for stochastic dominance test*

Description

Rejection cutoff value for stochastic dominance value based on alpha level.

Usage

```
barrett_donald_cutoff(alpha)
```

Arguments

alpha	Desired alpha level for stochastic dominance test.

Value

Cutoff value for stochastic dominance test

Author(s)

Emily Hsiao

`barrett_donald_p` *Helper function for stochastic dominance test*

Description

Calculates p-value of the stochastic dominance test

Usage

```
barrett_donald_p(statistic)
```

Arguments

`statistic`	Test statistic calculated in stochastic dominance test

Value

p-value of the test statistic

Author(s)

Emily Hsiao

`calculate_bandwidth` *Helper function for kernel smoother*

Description

Calculates the appropriate bandwidth for Nadaraya-Watson kernel smoother.

Usage

```
calculate_bandwidth(s)
```

Arguments

`s`	Vector of surrogate values.

Value

Desired bandwidth for kernel smoother.

Author(s)

Emily Hsiao

`gaussian_kernel` *Helper function for kernel smoother*

Description

Gaussian kernel used for kernel smoother.

Usage

```
gaussian_kernel(x)
```

Arguments

`x`	x

Value

Gaussian kernel applied to x

Author(s)

Emily Hsiao

`modified_S_stat` *Helper function for non-negative residual treatment effect test.*

Description

Calculates the value of the test statistic in the NNR test.

Usage

```
modified_S_stat(mu0_hat, mu1_hat, s0, y0, s1, y1, grid_x, boot = FALSE)
```

Arguments

`mu0_hat`	Kernel-smoothed estimate of μ_0 function.
`mu1_hat`	Kernel-smoothed estimate of μ_1 function.
`s0`	Vector of surrogate values in control group.
`y0`	Vector of endpoint values in control group.
`s1`	Vector of surrogate values in treatment group.
`y1`	Vector of endpoing values in the treatment group.
`grid_x`	Values of s over which supremum is calculated.
`boot`	Whether this is a bootstrapped statistic or the test statistic.

Value

`s_hat`	Value of the test statistic
`sup`	Value of the supremum over grid_x

Author(s)

Emily Hsiao

`monotonicity_test` *Monotonicity test*

Description

Runs the test of monotonicity for a regression function.

Usage

```
monotonicity_test(X, Y, h = NA, m = 5, bootstrap_n = 100,
alpha = 0.05)
```

Arguments

`X`	Vector of X values.
`Y`	Vector of Y values.
`h`	Bandwidth for the kernel smoother.
`m`	Window size to calculate linear regression.
`bootstrap_n`	Desired number of bootstrap samples.
`alpha`	Desired alpha level of the test.

Value

`T_m_value`	Value of the test statistic.
`p_val`	p-value of test
`reject`	whether the test rejects the null
`T_m_samples`	Vector of bootstrapped values of test statistic

Author(s)

Emily Hsiao

References

Hall, Peter, and Nancy E. Heckman. "Testing for monotonicity of a regression mean by calibrating for linear functions." Annals of Statistics (2000): 20-39.

`nnr_test` *Non-negative residual treatment effect function*

Description

Runs the test of non-negative residual treatment effect given two sets of surrogate and primary endpoint values.

Usage

```
nnr_test(s0, y0, s1, y1, n_bootstrap = 200, alpha = 0.05)
```

Arguments

`s0`	Vector of surrogate values in control group.
`y0`	Vector of endpoint values in control group.
`s1`	Vector of surrogate values in treatment group.
`y1`	Vector of endpoint values in treatment group.
`n_bootstrap`	Desired number of bootstrap samples.
`alpha`	Desired alpha level of test.

Value

`p_value`	p-value of test.
`reject`	Whether the test rejects the null hypothesis.
`s_hat`	Calculated value of test statistic.
`s_vec`	Vector of bootstrapped values of test statistic.

Author(s)

Emily Hsiao

References

Hsiao et all 2024 (Under review)

`Q` *Helper function for monotonicity test*

Description

Helper function for monotonicity test; should not be called directly by user.

Usage

```
Q(r, s, X)
```

Arguments

`r`	Index to start summation.
`s`	Index to end summation.
`X`	Vector of X values over which to sum.

Value

Q

Author(s)

Emily Hsiao

References

Hall, Peter, and Nancy E. Heckman. "Testing for monotonicity of a regression mean by calibrating for linear functions." Annals of Statistics (2000): 20-39.

`S` *Helper function*

Description

Helper function for monotonicity test.

Usage

```
S(a, b, r, s, X, Y)
```

Arguments

`a`	Value of a (regression coefficient)
`b`	Value of b (regression coefficient)
`r`	Index to start summation
`s`	Index to end summation
`X`	Vector of X values
`Y`	Vector of Y values

Value

Mean-squared error

Author(s)

Emily Hsiao

`sd_test` *Stochastic dominance test function*

Description

Runs the test of stochastic dominance given two vectors of surrogate values.

Usage

```
sd_test(s0, s1, alpha = 0.05)
```

Arguments

`s0`	Vector of surrogate values in control group.
`s1`	Vector of surrogate values in treatment group.
`alpha`	Desired alpha level of hypothesis test.

Value

`s_hat`	Value of test statistic
`p.value`	p-value of test
`reject`	Rejection decision of test

Author(s)

Emily Hsiao

References

Barrett, Garry F., and Stephen G. Donald. "Consistent tests for stochastic dominance." Econometrica 71.1 (2003): 71-104.

`smoother_fitter` *Helper function*

Description

Nadaraya-Watson kernel smoother

Usage

```
smoother_fitter(X, Y, kernel = gaussian_kernel, h)
```

Arguments

`X`	Vector of X values
`Y`	Vector of Y values
`kernel`	Kernel to use in the kernel smoother; defaults to Gaussian kernel
`h`	Bandwidth

Value

Returns a function which is the smoothed function; input takes in an x value.

Author(s)

Emily Hsiao

`test_assumptions` *Test assumptions to prevent surrogate paradox*

Description

Tests the assumptions necessary to prevent the surrogate paradox: stochastic dominance of surrogate values in the treatment group over control group, monotonicity of the relationship between surrogate and primary endpoint in both treatment and control group, and non-negative residual treatment effect of the treatment group over the control group. For computational efficiency, Version 2.0 of this package uses the monotonicity_test function from the MonotonicityTest package.

Usage

```
test_assumptions(s0 = NULL, y0 = NULL, s1 = NULL, y1 = NULL, trim = 0.95,
alpha = 0.05, type = "all", all_results = TRUE, direction = "positive",
monotonicity_bootstrap_n = 100, nnr_bootstrap_n = 200)
```

Arguments

`s0`	Vector of surrogate values in control group.
`y0`	Vector of primary endpoint values in control group.
`s1`	Vector of surrogate values in treatment group.
`y1`	Vector of primary endpoint values in treatment group.
`trim`	Proportion of data to keep after trimming the outliers. Defaults to 95%. Trims data by sorting by surrogate value and removing (1 - trim)/2 % of the lowest and highest surrogate values with their corresponding primary endpoint values.
`alpha`	Desired alpha level of tests.
`type`	Type of test to run. Defaults to "all"; possible inputs are "sd" (stochastic dominance), "monotonicity" (monotonicity), and "nnr" (non-negative residual treatment effect).
`all_results`	TRUE or FALSE; return all outputs from hypothesis tests. Defaults to TRUE.
`direction`	Direction of the test. Defaults to "positive", which tests that the treatment group stochastically dominates the control group, that μ_0 and μ_1 are monotonically increasing, and that $\mu_0 \leq \mu_1 \forall s$. Parameter "negative" tests that the control group stochastically dominates the treatment group, that μ_0 and μ_1 are monotonically decreasing, and that $\mu_1 \leq \mu_0 \forall s$.
`monotonicity_bootstrap_n`	Number of bootstrap samples for monotonicity test.
`nnr_bootstrap_n`	Number of bootstrap samples for nnr test.

Value

`result`	Table or string of results of the tests
`sd_result`	Detailed results of stochastic dominance test; only returned if all_results is TRUE
`monotonicity0_result`	Detailed results of monotonicity test in control group; only returned if all_results is TRUE
`monotonicity1_result`	Detailed results of monotonicity test in treatment group; only returned if all_results is TRUE
`nnr_result`	Detailed results of nnr test; only returned if all_results is TRUE

Author(s)

Emily Hsiao

References

Barrett, Garry F., and Stephen G. Donald. "Consistent tests for stochastic dominance." Econometrica 71.1 (2003): 71-104.

Hall, Peter, and Nancy E. Heckman. "Testing for monotonicity of a regression mean by calibrating for linear functions." Annals of Statistics (2000): 20-39.

Hsiao, Tian, Parast. "Avoiding the Surrogate Paradox: An Empirical Framework for Assessing Assumptions." 2025 (Under Review)

Examples

```
m_c <- function(s) 1 + 2 * s
m_t <- function(s) 1 + 2 * s

s_c <- rnorm(100, 3, 1)
y_c <- sapply(s_c, function(s) rnorm(1, m_c(s), 1))
s_t <- rnorm(100, 3, 1)
y_t <- sapply(s_t, function(s) rnorm(1, m_t(s), 1))

test_assumptions(
s0 = s_c, y0 = y_t, s1 = s_t, y1 = y_t, type = "sd"
)


test_assumptions(
s0 = s_c, y0 = y_t, s1 = s_t, y1 = y_t, type = "all")
```

`T_m` *Helper function for monotonicity test*

Description

Calculates the test statistic

Usage

```
T_m(m, X, Y)
```

Arguments

`m`	m window size
`X`	Vector of X values
`Y`	Vector of Y values

Value

`stat`	Value of the test statistic
`stat_vals`	Vector of statistics before taking maximum
`b_vals`	Values of b
`Q_vals`	Values of Q

Author(s)

Emily Hsiao

Index